Network Design and Protection Using Network Coding

Salah A. Aly
Electrical & Computer Eng. Dept.
New Jersey Institute of Technology
salah@njit.edu

Ahmed E. Kamal
Electrical & Computer Eng. Dept.
Iowa State University
kamal@iastate.edu

Anwar I. Walid
Bell Labs, Alcatel-Lucent
Murry Hill, NJ
anwar@research.bell-labs.com

Abstract—Link and node failures are two common fundamental problems that affect operational networks. Hence, protection of communication networks against such failures is essential for maintaining network reliability and performance. Network protection codes (NPC) are proposed to protect operational networks against link and node failures. Furthermore, encoding and decoding operations of such codes are well developed over binary and finite fields. Finding network topologies, practical scenarios, and limits on graphs applicable for NPC are of interest. In this paper, we establish limits on network protection codes and investigate several network graphs where NPC can be deployed. Furthermore, we construct graphs with minimum number of edges suitable for network protection codes deployment.

I. INTRODUCTION

With the increase in the capacity of backbone networks, the failure of a single link or node can result in the loss of significant amounts of information, which may lead to loss of revenues or even catastrophic failures. Network connections are therefore provisioned with the property that they can survive such edge and node failures. Several techniques have been introduced in the literature to achieve such goal, where either extra resources are added or some of the available network resources are reserved as backup circuits. Recovery from failures is also required to be agile in order to minimize the network outage time. This recovery usually involves two steps: fault diagnosis and location, and rerouting connections. Hence, the optimal network survivability problem is a multi-objective problem in terms of resource efficiency, operation cost, and agility [10], [11].

Allowing network relay nodes to encode packets is a novel approach that attracted much research work from both academia and industry with applications to enterprise networks, wireless communication and storage systems. This approach, which is known as network coding, offers benefits such as minimizing network delay, maximizing network capacity and enabling security and protection services, see [4], [9] and references therein. Network coding allows the sender nodes to combine/encode the incoming packets into one outgoing packet. Furthermore, the receiver nodes are allowed to decode those packets once they receive enough number of combinations. However, finding practical network topologies where

network coding can be deployed is a challenging problem. In order to apply network coding on a network with a large number of nodes, one must ensure that the encoding and decoding operations are done correctly over binary and finite fields.

There have been several applications for the edge disjoint paths (EDP) and node disjoint paths (NDP) problems in the literature including network flow, load balancing, routing, and optimal networks. In both cases (edge and vertex disjointness paths), deciding whether the pairs can be disjointedly connected is NP-complete [8].

A network protection scheme against a single link failure using network coding and reduced capacity is shown in [1]. The scheme is extended to protect against multiple link failures as well as against a single node failure. A protection scheme protects the communication links and network traffic between a group of senders and receivers in a large network with several relay nodes. This scheme is based on what we call *Network Protection Codes* (NPCs), which are defined in Section II. The encoding and decoding operations of such codes are defined in the case of binary and finite fields in [1], [2]. In this paper, we establish limits on network protection codes and investigate several network graphs where NPC can be deployed. In addition, we construct graphs with minimum number of edges valid for NPC deployment.

This paper is organized as follows. In Sections II and VI we present the network model and essential definitions. In Section III, we derive bounds on the minimum number of edges of graphs for NPC, and construct graphs that meet these bounds in Section V. Section IV presents limits on certain graphs that are applicable for NPC deployment.

II. NETWORK MODEL AND NPC DEFINITION

In this section we present the network model, define briefly network protection codes, and then state the problem. Further details can be found in [1].

A. Network Model

The network model is described as follows:

i) Let $\mathcal{N}$ be a network represented by an abstract graph $G = (\mathbf{V}, E)$, where $\mathbf{V}$ is the set of nodes and E be set of undirected edges. Let S and R are sets of independent
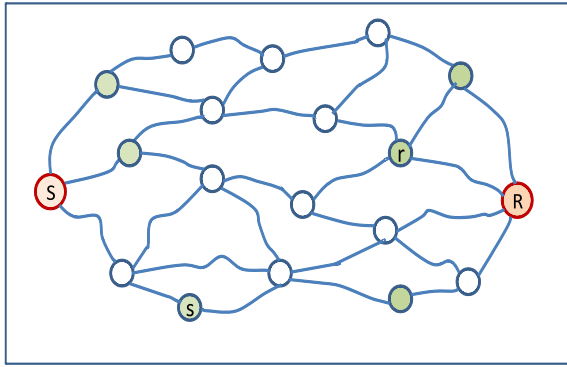
Fig. 1. A network model with a super source S and super receiver R. A set of sources and a set of receivers are shown in green color. We assume the source(s) and receiver(s) share k edge disjoint paths.

sources and destinations, respectively. The set $\mathbf{V} = V \cup S \cup R$ contains the relay, source, and destination nodes.

ii) The node can be a router, switch, or an end terminal depending on the network model $\mathcal{N}$ and the transmission layer.

iii) L is a set of links $L = \{L_1, L_2, \ldots, L_k\}$ carrying the data from the sources to the receivers. All connections have the same bandwidth, otherwise a connection with high bandwidth can be divided into multiple connections, each of which has a unit capacity. There are exactly k connections. For simplicity, we assume that the number of sources is less than or equal to the number of links. A sender with a high capacity can divide its capacity into multiple unit capacity, each of which has its own link. Put differently,

$$L_i = \{(s_i, w_{1i}), (w_{1i}, w_{2i}), \ldots, (w_{(\lambda)i}, r_i)\}, \qquad (1)$$

where $1 \leq i \leq k$ and $(w_{(j-1)i}, w_{ji}) \in E$, for some integer $\lambda \geq 1$.

iv) The failure on a link L_i may happen due to the network circumstance such as a link replacement, overhead, etc. We assume that the receiver is able to detect a failure and using the protection strategy is able to recover it.

B. NPC Definition

Let us assume a network model $\mathcal{N}$ with $t > 1$ path failures in the k working paths, i.e., paths carrying data from source(s) to receiver(s) [1]. One can define a *network protection code* NPC which protects k edge disjoint links as shown in the systematic matrix G in (2). In general, the systematic matrix G defines the source nodes that will send encoded messages and source nodes that will send only plain message without encoding. In order to protect k working paths, $k-t$ connections must carry plain data, and t connections must carry encoded data.

The generator matrix of the NPC for multiple link failures is given by:

$$G = \begin{bmatrix} 1 & 0 & \ldots & 0 & p_{11} & p_{12} & \ldots & p_{1t} \\ 0 & 1 & \ldots & 0 & p_{21} & p_{22} & \ldots & p_{2t} \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & p_{k-t,1} & p_{k-t,2} & \ldots & p_{k-t,t} \\ & \underbrace{\text{ident. } I_{k-t \times\ k-t}} & & & & \underbrace{\text{Submatrix } P_{k-t \times t}} & & \end{bmatrix}, \quad (2)$$

where $p_{ij} \in \mathbf{F}_q$, where $q \geq k - t + 1$, see [1].

The matrix G can be rewritten as

$$G = \begin{bmatrix} I_{k-t} & | & \mathbf{P}_{k-t,t} \end{bmatrix}, \qquad (3)$$

where $\mathbf{P}$ is the sub-matrix that defines the redundant data $\sum_{i=1}^{k-t} p_{ij}$ to be sent to a set of sources for the purpose of protection from multiple link failures, $1 \leq j \leq t$. The matrix G is defined explicitly using MDS optimal codes such as Reed-Solomon codes [6], [7]. Based on the above matrix, every source s_i sends its own message x_i to the receiver r_i via the link L_i. In addition t edge disjoint paths out of the k edge disjoint paths will carry encoded data.

Definition 1: An $[k, k-t]_q$ *network protection code* (NPC) is a k-t dimensional subspace of the space $\mathbf{F}_q^k$ that is able to recover from t edge disjoint path failures. The code protects k working paths and is defined by the matrix G described in Equation 2.

We say that a Network Protection Code (NPC) is feasible/valid on a graph G if the encoding and decoding operations can be achieved over the binary field $\mathbf{F}_2$ or a finite field with q elements $\mathbf{F}_q$ [1]. Also, we ensure that the set of senders (receivers) are connected with each other. We define the feasibility conditions of NPC, we will look for graphs that satisfy these conditions

Definition 2 (NPC Feasibility (validity)): Let S and R be sets of source(s) and receiver(s) in a graph G, as shown in Fig. 1. We say that the network protection code (NPC) is feasible (valid) for k edge disjoint connections (paths) from s_i in S to r_i in R, for $i = 1, 2, .., k$ if

(i) between any two sources s_i and s_j in S, there is a walk (path) $s_i \rightarrow s_j$. This means that the nodes in S share a tree.

(ii) between any two receivers r_i and r_j in R, there is a walk (path) $r_i \rightarrow r_j$. This means that the nodes in R share a tree.

(iii) there are k edge disjoint paths from S to R, the pairs $\langle s_i, r_i \rangle$ are different edge disjoint paths for all $1 \leq i \leq k$.

Therefore we say the graph G is valid for NPC deployment. By Definition 2, there are k edge disjoint paths in the graph from a set of k senders to a set of k receivers. This also include the case in which a single source sends k different messages through k edge disjoint paths to k receivers and vice versa. The feasibility of NPC guarantees that the encoding operations at the senders and decoding operations at the receivers can be achieved precisely.

C. Problem Statement

The max edge-disjoint paths (EDP) problem can be defined as follows. Let $G = (V, E)$ be an undirected graph represented by a set of nodes (network switches, routers, hosts, etc.), and a set of edges (network links, hops, single connection, etc.). Assume all edges have the same unit distance, and they are alike regarding the type of connection that they represent. A path from a source node u to a destination node v in V is represented by a set of edges in E. Put differently,

$$\langle u, v \rangle = \{(u, w_1), (w_1, w_2), \ldots, (w_j, v) \mid$$
$$u, v, w_i \in V, \quad (w_i, w_{i+1}) \in E\}. \quad (4)$$

Problem 1. Given k senders and k receivers. We can define a commodity problem, aka, k edge disjoint paths as follows. Given a network with a set of nodes V, and a set of links E, provision the edge disjoint paths to guarantee the encoding and decoding operations of NPC.

 i) Provision the k edge disjoint paths in G as

$$\begin{aligned} L &= \{L_1, L_2, \ldots, L_k\} \\ &= \{\langle s_j, r_j \rangle \mid \forall \ j = 1, \ldots, k, \ s_i \neq r_i \in V\} \end{aligned} \quad (5)$$

 be a set of commodities.

 ii) The set of sources $S = \{s_1, \ldots, s_k\}$ are connected with each other, as well as the set of receivers $R = \{r_1, \ldots, r_k\}$ are connected with each other as shown in Definition 2.

L are realizable in G if there exists mutually edge-disjoint connections from s_i to r_i for all $i = 1, \ldots, k$. Finding the set L in a given arbitrary graph G is an NP-complete problem as it is similar to edge-disjoint Menger's Problem and unsplittable flow [3].

Problem 2. Given positive integers n and k, and an NPC with k working paths, find a k-connected n-vertex graph G having the smallest possible number of edges. This graph by construction must have k edge disjoint paths and represents a network which satisfies NPC. This problem will be address in Section V.

III. NPC AND OPTIMAL GRAPH CONSTRUCTION WITH MINIMUM EDGES

One might ask what is the minimum number of edges on graphs, in which Network Protection Codes (NPC) is feasible/valid as shown in definition 2. We will answer this question in two cases: (i) the set of sources and receivers are predetermined (preselected from the network nodes). (ii) the sources and receivers are chosen arbitrary.

A. Single Source and Multiple Receivers

We consider the case of a single source and multiple receivers in an arbitrary graph with n nodes.

Lemma 3: Let G be a connected graph representing a network with n total nodes, among them a single source node and k receiver nodes. Assume a NPC from the source node to

the multiple receivers is applied. Then the minimum number of edges required to construct the graph G is given by

$$n + k - 2. \quad (6)$$

Proof: The graph G contains a single source, k receiver nodes, and $n - k - 1$ relay nodes (nodes that are not sources or receivers). To apply NPC, we must have k edge disjoint paths from the source to the k receivers. Also, all receivers must be connected by a tree with a minimum of $k - 1$ edges. The remaining $n - k - 1$ relay nodes in G can be connected with at least $n - k - 1$ edges. Therefore, the minimum number of nodes required to construct the graph G is given by

$$k + (k - 1) + (n - k - 1). \quad (7)$$

∎

B. Multiple Sources and Multiple Receivers

Lemma 4: Let G be a connected graph with n nodes and predetermined k sources and k receivers. Then the minimum number of edges required for predetermined k edge-disjoint paths for a feasible NPC solution on G is given by

$$E_{min} = n + k - 2 \quad (8)$$

Proof: We proceed the proof by constructing the graph G with a total number of nodes n and k sources (receivers).

 i) There are k sources that need a $k - 1$ edges represented by a tree. There are k receivers that need a $k - 1$ edges represented by a tree.

 ii) Assume every source node s_i is connected with a receiver node r_i has an l_i nodes in between for all $1 \leq i \leq k$. Therefore there are there are $l_i + 1$ edges in every edge-disjoint path, hence the number of edges from the sources to receivers is given by $k + \sum_{i=1}^{k} l_i$.

 iii) Assume an arbitrary node u exists in the graph G, then this node can be connected to a source (receiver) node or to another relay node. In either case, one edge is required to connected this node u to at least one node in G. Hence, number of edges required for all other relay nodes is given by $n - (2k + \sum_{i=1}^{k} l_i)$

 iv) Therefore the total number of edges is given by

$$\begin{aligned} E_{min} &= 2(k-1) + \left(k + \sum_{i=1}^{k} l_i\right) + \left(n - (k + \sum_{i=1}^{k} l_i)\right) \\ &= n + k - 2 \end{aligned} \quad (9)$$

∎

In the previous Lemma, we assume that the k sources and k receivers can be predetermined to minimize the number of edges on G. In the following Lemma, we assume that the sources and receivers can be chosen arbitrary among the n nodes of G.

Lemma 5: Let $G = (V, E)$ be a connected graph with n nodes and arbitrary chosen k sources and k receivers. Then the minimum number of edges required for any k edge-disjoint paths for a feasible NPC solution on G is given by

$$E_{min} = \lceil n(n - k + 1)/2 \rceil \quad (10)$$

Proof: In general, assume there are k connection paths, and the source and destination nodes do not share direction connections. In this case, every source node s_i in V must be connected to some relay nodes which are not receivers (destinations). Therefore, every source node must have a node degree of $(n-k+1)$. This agrement is also valid for any receiver node r_i in V. If we consider all $n = |V|$ nodes in the graph G, hence the total minimum number of edges E must be:

$$\lceil n(n-k+1)/2 \rceil \tag{11}$$

The ceiling value comes from the fact that both n and $(n-k+1)$ should not be odd. ∎

IV. EDGE DISJOINT PATHS IN k-CONNECTED AND REGULAR GRAPHS

In this section we look for certain graphs where NPC as feasible as shown in Definition 2. We will consider two cases: single source single receiver and single source multiple receivers. We will first consider the k-connected graphs as defined in Section VI. We derive bounds on the cases of $\kappa_e(G)$ and $\kappa_v(G)$ connectivity in a k-connected graph G.

A. Single Source to Single and Multiple Receivers

Whitney showed that the k-connected graph must have k edge disjoint paths between any two pair of nodes as shown in the following Theorem [5, Theorem 5.3.6].

Theorem 6 (Whitney 1932): A nontrivial graph G is k-connected if and only if for each pair u, v of vertices, there are at least k internally edge disjoint $\langle u, v \rangle$ paths in G.

Theorem 6 establishes conditions for k edge disjoint paths in a k-connected graph G. In order to make NPC feasible in a k-connected graph G, we ask for more two conditions: all receivers are connected with each others, as well as all source(s) are connected with each other.

Lemma 7: Let G be a non-trivial graph with a source node s and receiver node r. Then the NPC has a feasible solution with at least k edge disjoint paths **if and only if** G is a k-edge connected graph.

Proof: First, we know that if G is k-edge connected, then for each pair s, r of vertices, the degree of each node must be at least k. If not, then removing any number of edges less than k will disconnect the graph, and this contradicts the k-edge connectivity assumption. Each node connected with s will be a starting path to r or to another node in the graph. Consequently, every node must have a degree of at least k, and must have a path to r. Therefore, there are at least k internally edge disjoint $s-r$ paths in G. Hence, NPC is feasible by considering at least k edge disjoint paths.

Assume that NPC has a feasible/valid solution for $k' \geq k$, then there must exist $k' \geq k$ edge disjoint paths in G. Then for each pair of vertices s and r, there are at least k internally edge disjoint paths, then $\kappa_s(G), \kappa_r(G) \geq k$, for each s and r non-adjacent nodes. Therefore, the graph G is k-connected. ∎

Let s be a source in the network model $\mathcal{N}$ that sends k different source data to k receivers denoted by R. We need to
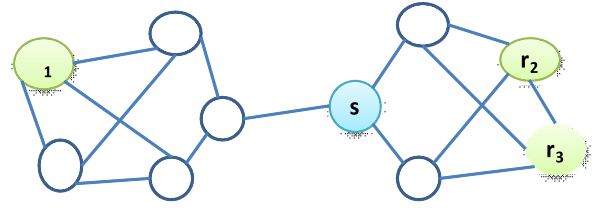

Fig. 2. An example of a regular graph of node degree three, in which NPC is not feasible/valid from a source node s to receiver ndoes r_1, r_2 and r_3. The receivers do not share a tree after removing the node s.

infer conditions for all R receivers to be connected with each other, and there are k edge disjoint paths from s to R. In this case, NPC will be feasible in the abstract graph G representing the network $\mathcal{N}$.

Theorem 8: Let G be a k-edge connected graph with a Hamiltonian cycle, and $s, r_1, r_2, \ldots, r_k$ be any $k+1$ distinct nodes in G. Then

i) there is a path L_i from s to r_i, for $i = 1, \ldots, k$, such that the collection $\{L_1, L_2, \ldots, L_k\}$ are internally edge disjoint paths.

ii) the nodes in the set R are connected with each other.

Therefore, NPC is feasible in the graph G.

The second condition ensures that there exists a tree in the graph G which connects all nodes in R without repeating edges from the k edge disjoint paths from s to R.

B. Regular Graphs and NPC Feasibility

We look to establish conditions on regular graphs to be feasible to apply NPC.

Theorem 9: Let G be a regular graph with minimum degree k. Then G has a k edge disjoint paths if and only if the min-cut separating a source from a sink is of at least k.

As shown in Fig. 2, the degree of each node is three and the min. cut separating the source node s from the receivers R is also three. However, we have the following negative result about NPC feasibility in regular graphs.

Lemma 10: There are regular graphs with node degree k, in which NPC is not feasible.

Proof: A certain example to prove this lemma would be a graph of 10 nodes, each of degree three, separated into two equally components connected with an edge, see Fig. 2. ∎

V. GRAPH CONSTRUCTION

We will construct graphs with a minimum number of edges for given certain number of vertices n and edge disjoint paths (connections) k, in which NPC can be deployed. Let $h_k(n)$ denote the minimum number of edges that a k-connected graph on n vertices must have. It is shown by F. Harary in 1962 that one can construct a k-connected graph $H_{k,n}$ on n vertices that has exactly $\lceil \frac{kn}{2} \rceil$ edges for $k \geq 2$. The construction begins with an n-cycle graph, whose vertices are consecutively numbered $0, 1, 2, \ldots, n-1$ clockwise. The proof of the following Lemma is shown in [5, Proposition 5.2.5.].

Input: Two positive integers k and n, number of connections and vertices, such that $k < n$.
Output: An optimal $H_{k,n}$ Harary graph for NPC.
Scatter the n isolated nodes ;
Let $r = \lfloor k/2 \rfloor$. ;
The construction of $H_{2r,n}$ Harary graph.;
foreach *i=0 to n-2* **do**
  foreach *j=i+1 to n-1* **do**
    if $j - i \leq r$ *OR* $n + i - j \leq r$ **then**
    Create an edge between vertices i and j.
  end
end
if k *is even* **then**
  Return graph H for NPC.
else
  if *n is even* **then**
    foreach *i=0 to* $\frac{n}{2} - 1$ **do**
      Create an edge between vertex i and vertices $i + \frac{n}{2}$
    end
  else
    Create an edge from vertex 0 to vertex $\frac{n-1}{2}$;
    Create an edge from vertex 0 to vertex $\frac{n+1}{2}$;
    foreach *i=1 to* $\frac{n-3}{2}$ **do**
      Create an edge between vertex i and vertex $i + \frac{n+1}{2}$
    end
  end
  Return graph H for NPC
end

Algorithm 1: Construction of an optimal graph for NPC with given n vertices, k connections and minimum number of edges.

Lemma 11: Let G be a k-connected graph on n nodes. Then the number of edges in G is at least $\lceil \frac{kn}{2} \rceil$. That is, $h_k(n) \geq \lceil \frac{kn}{2} \rceil$.

From Algorithm 1, one can ensure that there are k edge disjoint paths between any two nodes (one is sender and one is receiver). In addition, there are k edge disjoint paths from any node, which acts as a source, and k different nodes, which act as receivers. All nodes are connected together with a loop. Therefore, NPC can be deployed to such graphs.

Due to the fact that Harary's graph is k-connected [5, Theorem 5.2.6.], then using our previous result we can deduce that NPC is feasible for such graphs. Harary's graphs are optimal for NPC construction since they are k-edge connected graphs with the fewest possible number of edges.

VI. Conclusion

We proposed a method for optimal graph construction of network protection codes, and derived bounds on the minimum number of edges required for such graphs. We showered a method to construct optimal network graphs with minimum number of edges based on Harary graph constructions.

References

[1] S. A. Aly and A. E. Kamal. Network protection codes: Providing selfhealing in autonomic networks using network coding. *Submitted for publication*, 2009. arXiv:0812.0972v1 [cs.NI].

[2] S. A. Aly and A. E. Kamal. Network protection codes against link failures using network coding. In *Proc. IEEE GlobelComm 08, New Orleans, LA*, December 1-4, 2008. arXiv:0809.1258v1 [cs.IT].

[3] M. Blesa and C. Blum. Ant colony optimization from the maximum edge-disjoint paths problem. *EvoWorkshops 2004, LNCS 3005*, Springer- Verlag, pages 160169, 2004.

[4] C. Fragouli and E. Soljanin. Network coding applications, foundations and trends in networking. Hanover, MA, Publishers Inc., vol. 2, no. 2, pp. 135-269, 2007.

[5] J. Gross and J. Yellen. Graph Theory and its Applications. CRC Press, 1999.

[6] W. C. Huffman and V. Pless. Fundamentals of Error-Correcting Codes. Cambridge University Press, Cambridge, 2003.

[7] F.J. MacWilliams and N.J.A. Sloane. The Theory of Error-Correcting Codes. Amsterdam: North-Holland, 1977.

[8] J. Vygen. NP-completeness of some edge-disjoint paths problems. *Discrete Appl. Math.*, 61:8390, 1995.

[9] R. W. Yeung, S.-Y. R. Li, N. Cai, and Z. Zhang. Network Coding Theory. Now Publishers Inc., 2006.

[10] H. Zeng and A. Vukovic. The variant cycle-cover problem in fault detection and localization for mesh all-optical networks. *Photo Network communication*, 14:111122, 2007.

[11] H. Zhang, K. Zhu, and B. Mukkerjee. Backup reprovisioning to remedy the effect of multiple link failures in wdm mesh networks. *IEEE Journal on Selected Areas in Communications*, 24:5767, August 2006.

Appendix
Essential Definitions

We will proceed with some essential definitions. We assume that all graphs stated in this paper are undirected (bidirectional edges) unless stated otherwise. We define the edge-connectivity and node-connectivity of a graph G as follows.

Definition 12 (Edge-cut and node-cut): Given an undirected connected graph G, an edge-cut in a graph G is a set of edges such that its removal disconnects the graph. A node-cut in G is a set of nodes such that its removal disconnects the graph.

Definition 13 (node-edge-connectivity): The edge-connectivity of a connected graph G, denoted $\kappa_e(G)$ is the size of a smallest edge-cut. Also, the node-connectivity of a connected graph G, denoted $\kappa_v(G)$ is the minimum number of vertices whose removal can either disconnect the graph G or reduce it to a one-node graph.

The connectivity measures $\kappa_v(G)$ and $\kappa_e(G)$ are used in a quantified model of network survivability, which is the capacity of a network to retain connections among its nodes after some edges or nodes are removed.

Definition 14 (k-connected graph): a graph G is k-node connected if G is connected and $\kappa_v(G) \geq k$. Also, a graph G is k-edge connected if G is connected and every edge-cut has at least k edges, $\kappa_e(G) \geq k$.

We define two internal connections (paths) between nodes u and v in a graph G to be internally edge disjoint if they have no edge in common. This is also different from the node disjoint paths, in which no two disjoint paths share a common node. Throughout this paper, a path in a graph G does not contain the same node or edge twice, i.e a path $\langle u, v \rangle$ from a starting node u to an ending node v is a walk in a graph G [5].